What is claimed is:

1. In a system comprising an application, a framework, and an implementation class which provides an implementation for a particular service, a method performed by the framework, comprising:

receiving a request from an application for a customized implementation of a particular service;

instantiating an implementation class which provides an implementation for the particular service to give rise to an implementation instance;

determining a set of zero or more restrictions to be imposed on said customized implementation;

instantiating a wrapper class to give rise to a wrapper instance, said wrapper instance comprising enforcement logic for enforcing said restrictions;

encapsulating said implementation instance and said restrictions within said wrapper instance; and

providing said wrapper instance to the application as said customized implementation.

2. The method of claim 1, wherein said wrapper instance is invocable by the application without further interaction with the framework.

3. The method of claim 1, wherein the implementation class provides an unrestricted implementation for the particular service.

4. The method of claim 3, wherein the particular service is an encryption/decryption service, and wherein the unrestricted implementation provided

by the implementation class is capable of using unlimited encryption/decryption parameters.

5. The method of claim 4, wherein the unrestricted implementation provided by the implementation class is capable of using encryption/decryption keys of any size.

6. The method of claim 1, wherein said enforcement logic enforces said restrictions on said implementation instance.

7. The method of claim 6, wherein said enforcement logic enforces said restrictions on said implementation instance by:

receiving a set of desired parameters from the application;

determining whether the desired parameters exceed said restrictions; and

in response to a determination that the desired parameters exceed said restrictions, preventing said implementation instance from operating.

8. The method of claim 7, wherein said enforcement logic is invoked upon initialization of said wrapper instance.

9. The method of claim 1, wherein the system further comprises an exemption mechanism class which provides an implementation for a particular exemption mechanism, and wherein said method further comprises:

instantiating the exemption mechanism class to give rise to an exemption mechanism instance; and

encapsulating said exemption mechanism instance within said wrapper instance.

10. The method of claim 9, wherein said enforcement logic is invoked upon initialization of said wrapper instance, and when invoked, said enforcement logic:

determines whether said exemption mechanism instance has been invoked; and

in response to a determination that said exemption mechanism instance has not been invoked, preventing said implementation instance from operating.

11. The method of claim 1, wherein said wrapper instance comprises one or more invocable methods, wherein said implementation instance comprises one or more invocable methods, and wherein encapsulating comprises:

mapping the one or more invocable methods of said wrapper instance to the one or more invocable methods of said implementation instance.

12. The method of claim 1, wherein instantiating the implementation class comprises:

determining whether the implementation class is authentic; and

in response to a determination that the implementation class is authentic, instantiating the implementation class to give rise to said implementation instance.

13. The method of claim 12, wherein the implementation class has a digital signature associated therewith, and wherein determining whether the implementation class is authentic comprises:

verifying said digital signature.

14. The method of claim 12, wherein the implementation class authenticates the framework prior to giving rise to said implementation instance.

15. The method of claim 1, wherein determining the set of zero or more restrictions comprises:

accessing information specifying one or more limitations; and

processing said limitations to derive said restrictions.

16. The method of claim 15, wherein the particular service is an encryption/decryption service, and wherein said information comprises a set of one or more default encryption limitations.

17. The method of claim 16, wherein said default encryption limitations are derived by merging multiple jurisdiction policies and extracting therefrom the most restrictive encryption limitations.

18. The method of claim 1, wherein determining the set of zero or more restrictions comprises:

accessing information specifying one or more limitations;

determining permissions, if any, granted to the application; and

reconciling said limitations and said permissions to derive said restrictions.

19. The method of claim 18, wherein said limitations and said permissions are reconciled to derive restrictions which are least restrictive.

20. The method of claim 18, wherein the particular service is an encryption/decryption service, and wherein said information comprises a set of one or more default encryption limitations, and a set of zero or more exempt encryption limitations which apply when one or more exemption mechanisms are implemented.

21. The method of claim 20, wherein said default encryption limitations and said exempt encryption limitations are derived by merging multiple jurisdiction policies and extracting therefrom the most restrictive encryption limitations.

22. The method of claim 20, wherein reconciling said limitations and said permissions comprises:

determining whether the application has been granted any permissions; and

in response to a determination that the application has not been granted any permissions, deriving said restrictions from said set of default encryption limitations.

23. The method of claim 20, wherein reconciling said limitations and said permissions comprises:

determining whether the application has been granted any permissions which require implementation of a particular exemption mechanism;

in response to a determination that the application has been granted a permission which requires implementation of a particular exemption mechanism, determining whether said exempt encryption limitations allow said particular exemption mechanism to be implemented; and

in response to a determination that said exempt encryption limitations allow said particular exemption mechanism to be implemented, deriving said restrictions from said set of exempt encryption limitations.

24. In a system comprising an application and an implementation class which provides an implementation for a particular service, a framework comprising:

a mechanism for receiving a request from an application for a customized implementation of a particular service;

a mechanism for instantiating an implementation class which provides an implementation for the particular service to give rise to an implementation instance;

a mechanism for determining a set of zero or more restrictions to be imposed on said customized implementation;

a mechanism for instantiating a wrapper class to give rise to a wrapper instance, said wrapper instance comprising enforcement logic for enforcing said restrictions;

a mechanism for encapsulating said implementation instance and said restrictions within said wrapper instance; and

a mechanism for providing said wrapper instance to the application as said customized implementation.

25. The framework of claim 24, wherein said wrapper instance is invocable by the application without further interaction with the framework.

26. The framework of claim 24, wherein the implementation class provides an unrestricted implementation for the particular service.

27. The framework of claim 26, wherein the particular service is an encryption/decryption service, and wherein the unrestricted implementation provided by the implementation class is capable of using unlimited encryption/decryption parameters.

28. The framework of claim 27, wherein the unrestricted implementation provided by the implementation class is capable of using encryption/decryption keys of any size.

29. The framework of claim 24, wherein said enforcement logic enforces said restrictions on said implementation instance.

30. The framework of claim 29, wherein said enforcement logic enforces said restrictions on said implementation instance by:

receiving a set of desired parameters from the application;

determining whether the desired parameters exceed said restrictions; and

in response to a determination that the desired parameters exceed said restrictions, preventing said implementation instance from operating.

31. The framework of claim 30, wherein said enforcement logic is invoked upon initialization of said wrapper instance.

32. The framework of claim 24, wherein the system further comprises an exemption mechanism class which provides an implementation for a particular exemption mechanism, and wherein said framework further comprises:

a mechanism for instantiating the exemption mechanism class to give rise to an exemption mechanism instance; and

a mechanism for encapsulating said exemption mechanism instance within said wrapper instance.

33. The framework of claim 32, wherein said enforcement logic is invoked upon initialization of said wrapper instance, and when invoked, said enforcement logic:

determines whether said exemption mechanism instance has been invoked; and

in response to a determination that said exemption mechanism instance has not been invoked, preventing said implementation instance from operating.

34. The framework of claim 24, wherein said wrapper instance comprises one or more invocable methods, wherein said implementation instance comprises one or more invocable methods, and wherein the mechanism for encapsulating comprises:

a mechanism for mapping the one or more invocable methods of said wrapper instance to the one or more invocable methods of said implementation instance.

35. The framework of claim 24, wherein the mechanism for instantiating the implementation class comprises:

a mechanism for determining whether the implementation class is authentic; and

a mechanism for instantiating, in response to a determination that the implementation class is authentic, the implementation class to give rise to said implementation instance.

36. The framework of claim 35, wherein the implementation class has a digital signature associated therewith, and wherein the mechanism for determining whether the implementation class is authentic comprises:

a mechanism for verifying said digital signature.

37. The framework of claim 35, wherein the implementation class authenticates the framework prior to giving rise to said implementation instance.

38. The framework of claim 24, wherein the mechanism for determining the set of zero or more restrictions comprises:

a mechanism for accessing information specifying one or more limitations; and

a mechanism for processing said limitations to derive said restrictions.

39. The framework of claim 38, wherein the particular service is an encryption/decryption service, and wherein said information comprises a set of one or more default encryption limitations.

40. The framework of claim 39, wherein said default encryption limitations are derived by merging multiple jurisdiction policies and extracting therefrom the most restrictive encryption limitations.

41. The framework of claim 24, wherein the mechanism for determining the set of zero or more restrictions comprises:

a mechanism for accessing information specifying one or more limitations;

a mechanism for determining permissions, if any, granted to the application; and

a mechanism for reconciling said limitations and said permissions to derive said restrictions.

42. The framework of claim 41, wherein said limitations and said permissions are reconciled to derive restrictions which are least restrictive.

43. The framework of claim 41, wherein the particular service is an encryption/decryption service, and wherein said information comprises a set of one or more default encryption limitations, and a set of zero or more exempt encryption limitations which apply when one or more exemption mechanisms are implemented.

44. The framework of claim 43, wherein said default encryption limitations and said exempt encryption limitations are derived by merging multiple jurisdiction policies and extracting therefrom the most restrictive encryption limitations.

45. The framework of claim 43, wherein the mechanism for reconciling said limitations and said permissions comprises:

a mechanism for determining whether the application has been granted any permissions; and

a mechanism for deriving, in response to a determination that the application has not been granted any permissions, said restrictions from said set of default encryption limitations.

46. The framework of claim 43, wherein the mechanism for reconciling said limitations and said permissions comprises:

a mechanism for determining whether the application has been granted any permissions which require implementation of a particular exemption mechanism;

a mechanism for determining, in response to a determination that the application has been granted a permission which requires implementation of a particular exemption mechanism, whether said exempt encryption limitations allow said particular exemption mechanism to be implemented; and

a mechanism for deriving, in response to a determination that said exempt encryption limitations allow said particular exemption mechanism to be implemented, said restrictions from said set of exempt encryption limitations.

47. In a system comprising an application and an implementation class which provides an implementation for a particular service, a computer readable medium having stored thereon instructions which, when executed by one or more processors, cause the one or more processors to implement a framework which

dynamically constructs a customized implementation, said computer readable medium comprising:

instructions for causing one or more processors to receive a request from an application for a customized implementation of a particular service;

instructions for causing one or more processors to instantiate an implementation class which provides an implementation for the particular service to give rise to an implementation instance;

instructions for causing one or more processors to determine a set of zero or more restrictions to be imposed on said customized implementation;

instructions for causing one or more processors to instantiate a wrapper class to give rise to a wrapper instance, said wrapper instance comprising enforcement logic for enforcing said restrictions;

instructions for causing one or more processors to encapsulate said implementation instance and said restrictions within said wrapper instance; and

instructions for causing one or more processors to provide said wrapper instance to the application as said customized implementation.

48. The computer readable medium of claim 47, wherein said wrapper instance is invocable by the application without further interaction with the framework.

49. The computer readable medium of claim 47, wherein the implementation class provides an unrestricted implementation for the particular service.

50. The computer readable medium of claim 49, wherein the particular service is an encryption/decryption service, and wherein the unrestricted implementation provided by the implementation class is capable of using unlimited encryption/decryption parameters.

51. The computer readable medium of claim 50, wherein the unrestricted implementation provided by the implementation class is capable of using encryption/decryption keys of any size.

52. The computer readable medium of claim 47, wherein said enforcement logic enforces said restrictions on said implementation instance.

53. The computer readable medium of claim 52, wherein said enforcement logic enforces said restrictions on said implementation instance by:

receiving a set of desired parameters from the application;

determining whether the desired parameters exceed said restrictions; and

in response to a determination that the desired parameters exceed said restrictions, preventing said implementation instance from operating.

54. The computer readable medium of claim 53, wherein said enforcement logic is invoked upon initialization of said wrapper instance.

55. The computer readable medium of claim 47, wherein the system further comprises an exemption mechanism class which provides an implementation

for a particular exemption mechanism, and wherein said computer readable medium further comprises:

instructions for causing one or more processors to instantiate the exemption mechanism class to give rise to an exemption mechanism instance; and

instructions for causing one or more processors to encapsulate said exemption mechanism instance within said wrapper instance.

56. The computer readable medium of claim 55, wherein said enforcement logic is invoked upon initialization of said wrapper instance, and when invoked, said enforcement logic:

determines whether said exemption mechanism instance has been invoked; and

in response to a determination that said exemption mechanism instance has not been invoked, preventing said implementation instance from operating.

57. The computer readable medium of claim 47, wherein said wrapper instance comprises one or more invocable methods, wherein said implementation instance comprises one or more invocable methods, and wherein the instructions for causing one or more processors to encapsulate comprises:

instructions for causing one or more processors to map the one or more invocable methods of said wrapper instance to the one or more invocable methods of said implementation instance.

58. The computer readable medium of claim 47, wherein the instructions for causing one or more processors to instantiate the implementation class comprises:

instructions for causing one or more processors to determine whether the implementation class is authentic; and

instructions for causing one or more processors to instantiate, in response to a determination that the implementation class is authentic, the implementation class to give rise to said implementation instance.

59. The computer readable medium of claim 58, wherein the implementation class has a digital signature associated therewith, and wherein the instructions for causing one or more processors to determine whether the implementation class is authentic comprises:

instructions for causing one or more processors to verify said digital signature.

60. The computer readable medium of claim 58, wherein the implementation class authenticates the framework prior to giving rise to said implementation instance.

61. The computer readable medium of claim 47, wherein the instructions for causing one or more processors to determine the set of zero or more restrictions comprises:

instructions for causing one or more processors to access information specifying one or more limitations; and

instructions for causing one or more processors to process said limitations to derive said restrictions.

62. The computer readable medium of claim 61, wherein the particular service is an encryption/decryption service, and wherein said information comprises a set of one or more default encryption limitations.

63. The computer readable medium of claim 62, wherein said default encryption limitations are derived by merging multiple jurisdiction policies and extracting therefrom the most restrictive encryption limitations.

64. The computer readable medium of claim 47, wherein the instructions for causing one or more processors to determine the set of zero or more restrictions comprises:

instructions for causing one or more processors to access information specifying one or more limitations;

instructions for causing one or more processors to determine permissions, if any, granted to the application; and

instructions for causing one or more processors to reconcile said limitations and said permissions to derive said restrictions.

65. The computer readable medium of claim 64, wherein said limitations and said permissions are reconciled to derive restrictions which are least restrictive.

66. The computer readable medium of claim 64, wherein the particular service is an encryption/decryption service, and wherein said information comprises a set of one or more default encryption limitations, and a set of zero or more exempt

encryption limitations which apply when one or more exemption mechanisms are implemented.

67. The computer readable medium of claim 66, wherein said default encryption limitations and said exempt encryption limitations are derived by merging multiple jurisdiction policies and extracting therefrom the most restrictive encryption limitations.

68. The computer readable medium of claim 66, wherein the instructions for causing one or more processors to reconcile said limitations and said permissions comprises:

instructions for causing one or more processors to determine whether the application has been granted any permissions; and

instructions for causing one or more processors to derive, in response to a determination that the application has not been granted any permissions, said restrictions from said set of default encryption limitations.

69. The computer readable medium of claim 66, wherein the instructions for causing one or more processors to reconcile said limitations and said permissions comprises:

instructions for causing one or more processors to determine whether the application has been granted any permissions which require implementation of a particular exemption mechanism;

instructions for causing one or more processors to determine, in response to a determination that the application has been granted a permission which requires

implementation of a particular exemption mechanism, whether said exempt encryption limitations allow said particular exemption mechanism to be implemented; and

instructions for causing one or more processors to derive, in response to a determination that said exempt encryption limitations allow said particular exemption mechanism to be implemented, said restrictions from said set of exempt encryption limitations.